

Name: \_\_\_\_\_

USN Number: \_\_\_\_\_

Mobile Number: \_\_\_\_\_

**SITETRACKER**

## INTERNSHIP PROGRAMMING EXERCISE (2025)

### General Instructions

1. This questionnaire consists of 2 problems and you are expected to solve both.
2. You have 60 Mins of time in total.
3. Use any OOPS language (c++, java, python, javascript) to write code.

### Problem 1: Merge Overlapping Meetings

**Description:** Given meeting time intervals, merge all overlapping intervals and output the consolidated schedule.

**Input Format:**

- First line: Integer n (number of meetings)
- Next n lines: Two space-separated integers start and end

**Output Format:**

- Each merged interval on a new line: start end (sorted by start time)

**Sample 1:**

Input:

```
4
1 5
3 7
8 10
9 12
```

Output:

```
1 7
8 12
```

**Sample 2:**

Input:

```
7
1 10
2 6
5 15
18 20
21 25
30 40
35 45
```

Output:

```
1 15
18 20
21 25
30 45
```

## Problem 2: Find Heavy Users

**Description:** Given bank transactions, find all users who made 3 or more transactions within any 60-second window.

### Input Format:

- First line: Integer  $n$  (transactions)
- Next  $n$  lines: time user\_id (space-separated, transactions sorted by time)

### Output Format:

- Space-separated list of user IDs (unique, sorted ascending)

### Sample 1:

Input:

```
6
1 101
30 101
59 101
65 101
100 102
101 102
```

Output:

```
101
```

Explanation: User 101 has transactions at times 1, 30, 59, 65.  
The 60-second window [5, 65] contains 3 transactions (30, 59, 65).

### Sample 2:

Input:

```
8
10 201
30 201
50 201
70 201
5 202
25 202
45 202
90 203
```

Output:

```
201 202
```

Explanation:

- User 201: 4 transactions at 10, 30, 50, 70 within [10,70] (60s window)
- User 202: 3 transactions at 5, 25, 45 within [5,65] (60s window)
- User 203: Only 1 transaction, not flagged